

ROS Basics

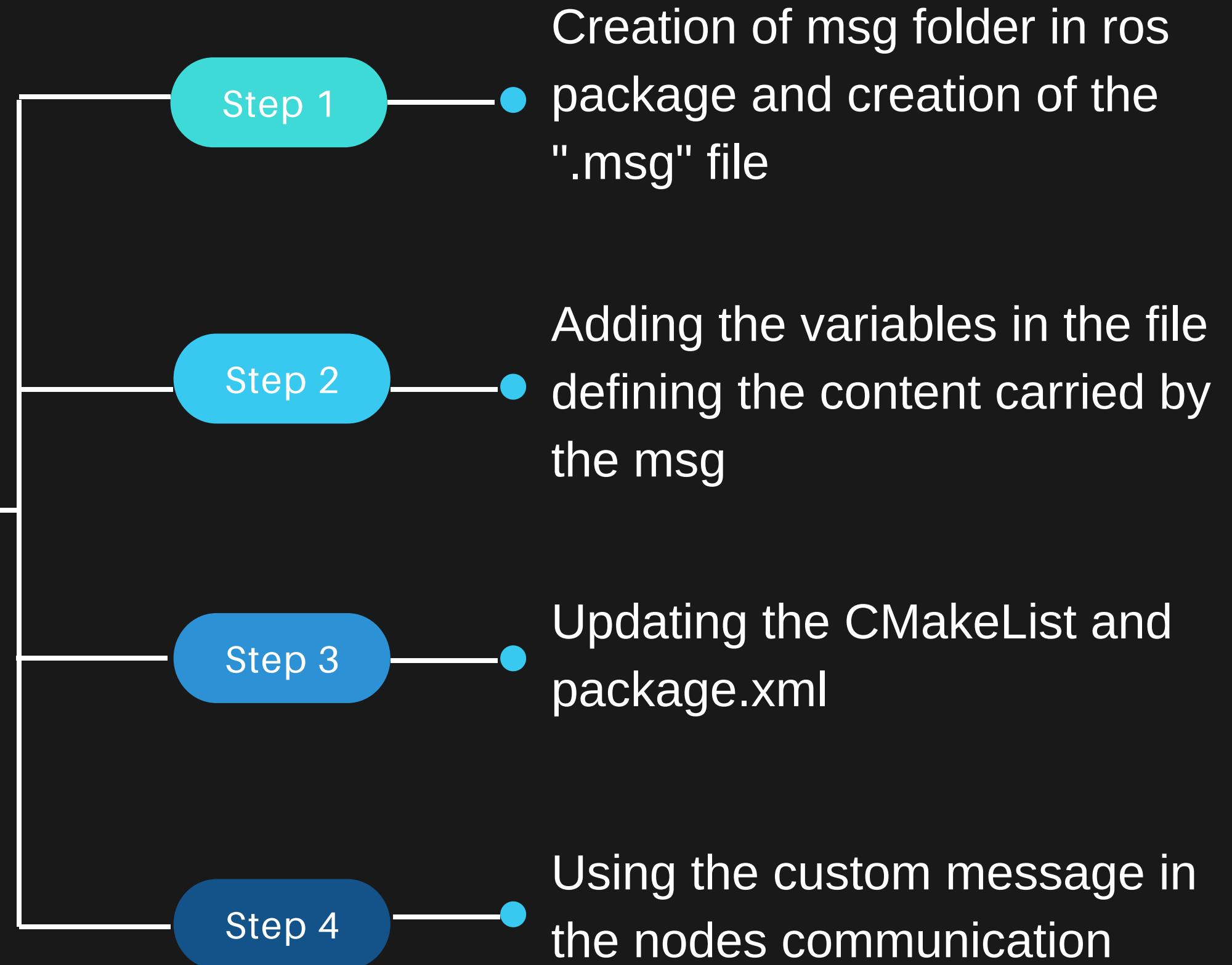
Topic Covered: ROS Custom Messages

Part 5

100Days of ROS series

By
M.N.Subhash
@malladisubhash

CREATION OF CUSTOM MESSAGES



Creation of the ROS message structure

- msg files are simple text files that describe the fields of the ROS message, it indicates the data a ROS message will carry
- The msg files are stored in the "msg " directory in the package
- The basic structure of the message file is
 1. It contains the field type and field name, which is the data the message can hold
 2. The fields that can be used are:
 - int8, int16, int32, int64 (plus uint*)
 - float32, float64
 - string
 - time, duration
 - other msg files
 - variable-length array[] and fixed-length array[C]

These indicate the type of data the ros message can carry

Adding the data to the MSG file

- According to the task 8 which deals with the basic communication b/w the nodes using custom msgs, the contents of the msg file are as follows

```
string name  
uint32 number
```

This message carries the information which contains the "name" data and "number" data

Changes to the CMakelist and Package file

- As the ROS needs to recognize the creation of the new custom message and also to convert the msg file to the source code the following changes in the **CMakelist.txt** file and the **Package.xml** file
- In the **package.xml** the following changes needs to be done

```
<build_depend>message_generation</build_depend>  
<build_depend>message_runtime</build_depend>  
  
<build_export_depend>message_runtime</build_export_depend>  
<build_export_depend>message_generation</build_export_depend>  
  
<exec_depend>message_runtime</exec_depend>
```

- Now the following changes need to be made to the **CMakeList.txt**
 1. In the file find the `find_package` and add the **message generation** part to it.
- Now search for the **add_message_files** block and add the name of the message file created

```
add_message_files(  
FILESType  
custommessage.msg //This is the custommessage file created as a part of the task  
)
```

- Uncomment the **generate_message** code block, this will generate the source files from the msg definition

These are the changes needs to be done to successfully create the ROS custom message

Source : <http://wiki.ros.org/ROS/Tutorials/CreatingMsgAndSrv>

Application of the custom ROS message

- To see the structure of the custom message or any message i.e the information ROS message. Can be seen by the following command

rosmsg show [message type]

- The application of the custom rosmessage is similar to using any ROS message. The publisher code is as follows

from task7_custom_messages.msg import custommessage //This is how the custommessage is instantiated in the code

//**custom_msg_info** in the topic to which data is published and **custom_msg_publisher** is the name of the publisher

```
cm_publisher = rospy.Publisher('custom_msg_info', custommessage, queue_size=10)
rospy.init_node('custom_msg_publisher', anonymous=False)
```

//The data which is sent through the channel

```
msg.name = "sensor1"
msg.number = 1.0
```

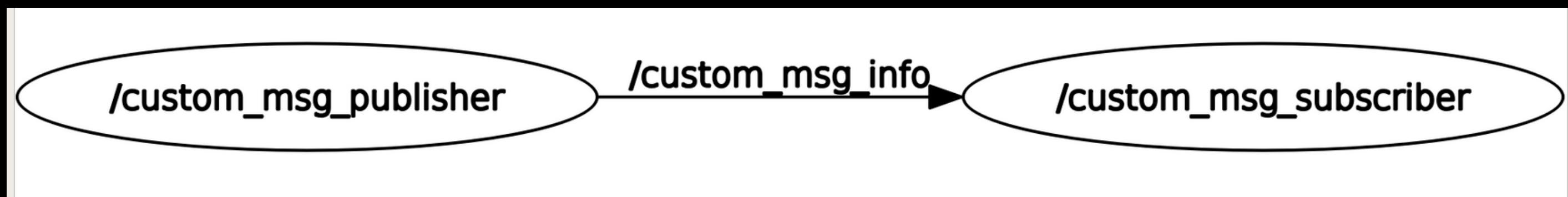
The subscriber code is as follows:

- Initially the custommessage is instantiated in the code and by using a callback function the data from the publisher is logged

rospy.loginfo('Distance reading from the %s is: %f',mymsg.name,mymsg.number)

remaining part of the similar to any standard subscriber node

- The rqt_graph is as follows



About Me

I am currently pursuing my degree at Sir M Visvesvaraya Institute of Technology, Bangalore.

The domains I am interested in are Electronics and Robotics.

My interest in Robotics started when I was in my second year, Currently improving my skills in ROS and took a 100daysofROS challenge to focus mainly on the hardware implementation of ROS

I put regular updates of my progression of challenges on Twitter
As a part of the challenge, I will be documenting the important things and will be making a concise version of them.

Follow me to catch up with ROS and learn how to implement it on hardware.

LinkedIn profile: linkedin.com/in/malladisubhash

Twitter Handle: [@mns2610](https://twitter.com/@mns2610)

GitHub ROS repository link: https://github.com/malladi2610/100_days_of_ROS